Association for Information Systems
AIS Electronic Library (AISeL)

8-25-1995

Application Portfolio Diversity and Software Maintenance Productivity: An Empirical Analysis

Rajiv D. Banker
University of Minnesota

Gordon B. Davis
University of Minnesota

Sandra A. Slaughter
Carnegie Mellon University

Follow this and additional works at: http://aisel.aisnet.org/amcis1995

Application Portfolio Diversity and Software Maintenance Productivity: An Empirical Analysis

Rajiv D. Banker
Gordon B. Davis
Carlson School of Management
University of Minnesota

Sandra A. Slaughter
Graduate School of Industrial Administration
Carnegie Mellon University

INTRODUCTION

The research addresses the issue of productivity in application software maintenance. Specifically, it examines the effect of diversity in tools, techniques, hardware and software associated with the portfolio being maintained. In manufacturing environments, there is some evidence to suggest that production of products where there is little sharing of inputs and production processes reduces focus and results in lower manufacturing performance (Skinner, 1974). In economics, it is argued that there are cost complementarities or economies of scope in sharing common inputs and processes among various products with commonalities in production, and diseconomies of scope when inputs and processes differ (Panzar and Willig, 1977, 1981).

In the software maintenance context, the issue of diversity and its effect on productivity is particularly salient. Software maintenance is work done to enhance software functionality, correct errors and improve the performance of software (Schneidewind, 1987). It is a costly activity for organizations, requiring from 50 to 80% of the Information Systems (IS) budget and representing more than threefourths of software costs on a life cycle basis (Arthur, 1988).

Application portfolio diversity, i.e., differences in technical platforms, software languages, and development tools and techniques in the set of the organization's software systems, arises as a consequence of the organization's information technology infrastructure decisions. To meet a particular customer need, an IS group acquires or develops software using a certain tool, methodology, and hardware platform. However, it may be that the software does not fit well into the organization's existing technical platform. Furthermore, the software may have been developed using a different methodology or tools than other software systems in the organization's portfolio. This diversity may have the result of increased difficulty in software maintenance because software enhancement can require modification of multiple software systems that have been created using a variety of languages, tools and techniques.

The results of our analysis suggest that software portfolio diversity reduces productivity in software maintenance. Potential inefficiencies from diversity in software maintenance can arise from several causes. Switching costs are incurred due to multiple, varied process flows and frequent change over in processes required when modifying software created using different methodologies and tools. Diversity may also increase the difficulty of software quality control, testing and verification; for example, inefficiencies may occur due to the complexities of conducting system and integration testing across multiple technical platforms. Finally, there may be costs due to the difficulties in selecting project team members with the multiple and varied skills required to modify diverse sets of software.

METHODOLOGY

To assess the implications of application portfolio diversity for software enhancement project efficiency, we analyzed 121 software enhancement projects completed at a large financial institution. The IS department for the organization is located at company headquarters and supports all centralized computer processing activities for the company. More than 90% of IS resources are devoted to supporting the organization's existing software portfolio. The organization's software portfolio exceeds 250,000 function points (150 million lines of code) in size, and ranges in vintage from 1969 to 1994. The portfolio is diverse in platform, languages, and development practices. The software executes on more than ten different hardware platforms, and has been created using more than twenty different design tools and methodologies and ten different software languages. Sixty percent of the software was developed inhouse; the remainder was purchased.

Data on the number of different design, development and testing tools and techniques, and hardware platforms used in software enhancement projects was obtained from the IS organization's archival records. A statistical regression model was constructed to determine the effects of project diversity on productivity. The model assesses the effects on project inefficiency of the number of hardware platforms impacted by the project, the number of tools and techniques used in project design, the number of tools and techniques used in project development, andthe number of tools and techniques used in project testing. A listing of the kinds of different platforms, tools and techniques for the projects is presented in Figure 1. Project inefficiency is measured by the reciprocal of the productivity rating calculated for the projects using Data Envelopment Analysis (DEA). DEA is a nonparametric methodology for production frontier estimation developed by Charnes, Cooper, and Rhodes (1981) and extended to a formal production economics framework by Banker, Charnes, and Cooper (1984). In our analysis, we employ the project efficiency rating determined by the Banker, Charnes and Cooper DEA methodology.

We estimated our model using Ordinary Least Squares (OLS), and conducted checks to ensure that the assumptions of OLS were satisfied for the estimation. These checks included Kolmogorov's test for normality of residuals (Greene, 1993), White's test for

homoscedasticity (White, 1980), and tests for multicollinearity and influence of outliers (Belsley, Kuh and Welsch, 1980).

RESULTS

Figure 1 - Types of Project Platforms, Tools, Techniques Used at Data Site

Hardware Platforms: ATM's, IBM Mainframe, PC PC-LAN, Stratus, System 3X, Tandem, IBM 4700 Financial System Hardware.
Software Design Tools: Bachman, Excelerator, HPS CASE Tool, Pacbase, PacBench, Pacdesign, Flowchart2, SDF2.
Software Design Techniques: Data Normalizaton, Data Flow Diagrams, Entity Relationship Models, Information Engineering, Holland Proplanner, Structured Walkthroughs, SISP/Holland, Flowcharts, Merise Systems Analysis.
Software Development Tools: CICS, CLISTS, Fileaid, Flowcharting iii+, HPS CASE Tool, ISPF, Optimizer, Panvalet, TSO/Editor, WSF2, Pacbase, Pacbench, Pacdesign, Pactable.
Software Development Techniques (including software languages and file types): BAL, Base24, BDAM, BSAM, C, COBOL, COBOL2, DB2, Dbase2, DBase4, Focus, IMS, KBMS, MF COBOL, PL1, QDAM, RPG, SAS, VSAM.
Testing Tools: Abend-Aid, CEDF, Comparex, DADS, DBUG-Aid, File-Aid, IBM Utilities, IMS-Xpert, ISPF, ISPF Dialogue Test Facilities, JCL Check, QMF/SPUFI, TPNS, TRAPS, WSF2, Xpediter.
Testing Techniques: Acceptance, Integration, System, Unit, Parallel, Regression, Stress, Volume.

Figure 2 - Enhancement Project Profile

Measure	Mean	SDev	Min	Max
# platforms	1.46	0.86	0.0	4.0
# des tools	0.02	0.16	0.0	1.0
# des techs	0.46	0.79	0.0	4.0
# dev tools	3.77	1.97	0.0	6.0
# dev techs:	3.45	2.02	0.0	9.0
# languages	1.77	0.88	1.0	4.0
# file types	1.43	0.97	0.0	4.0
# test tools	4.78	2.76	0.0	12.0
# test techs	2.68	2.36	0.0	8.0
proj ineff*	5.29	2.17	1.0	13.0 *(1 implies low and 12 high inefficiency.)

Figure 3 - Correlation Matrix

	ProjFP	TExp	HWplt	Dstool	Dstech	Dvtool	Dvtech	Ttool	Ttech
ProjFP	1.000								

```
TeamExp            0.102   1.000
HWplat     0.003   -.081   1.000
Destool    0.101   0.048   0.416*  1.000
Destech    -.048   0.105   0.287*  0.255*  1.000
Devtool    0.114   0.223   -.053   0.015   0.206*  1.000
Devtech    -/048   0.105   0.287*  0.255*  0.194*  0.120   1.000
Testtool   0.029   0.227*  0.078   0.148   0.303*  0.500*  0.391*  1.000
Testtech   -.003   -.121   0.278*  0.270*  0.499*  -.058   0.191   0.311*  1.000
```

Figure 4 - Regression Results

Dep Var = ProjIneff. Results: R-squared=.893; adjusted R squared=.880; F-test (model)=68.703 (p<.0001)

Variable	Coeff.	ParamEst	T-value	[* indicates significance at 5% level]
Intercept	B0	0.924628	3.857*	
TeamExp	B1	-.094017	-1.186	
HWplat	B2	0.372831	3.896*	
Destool	B3	2.130279	4.043*	
Destech	B4	0.337680	2.664*	
Devtool	B5	0.548611	5.925*	
Devtech	B6	0.273076	6.196*	
Testtool	B7	0.372389	10.695*	
Testtech	B8	0.028282	0.689	

Figure 2 presents a profile of the 121 software enhancement projects included in this analysis. On average, the typical enhancement project impacted 1.5 different hardware platforms, used 3.5 different kinds of development tools including 1.8 different software languages and 1.5 different file types, 3.8 different kinds of development tools, 4.8 different kinds of testing tools, and 2.7 different kinds of testing techniques. The correlation matrix presented in Figure 3 suggests that this diversity is not related to project size as measured by project function points, and that correlations between the tools and techniques used in each phase are modest. The regression model in Figure 4 indicates that diversity in the number of tools and techniques is significantly associated with increased inefficiency in enhancement projects. Of particular import is the number of testing tools employed in the projects.

IMPLICATIONS AND CONTRIBUTIONS

Our results suggest that software portfolio diversity is strongly associated with inefficiencies in software enhancement at our data site, particularly in software testing. These results have implications for both practice and research in information technology management. Specifically, they imply that portfolio diversity should be an important

software life cycle consideration, because the costs of diversity emerge in software maintenance.

SELECTED REFERENCES

Banker, R.D., A. Charnes, and W.W. Cooper, "Some Models for Estimating Technical and Scale Inefficiencies in DEA," Management Science, 30:9, September 1984, p. 10781092.

Panzar, J.C., and R.D. Willig, "Economies of Scale in Multioutput Production," Quarterly Journal of Economics, vol. 91, 1977, p. 481493.

Panzar, J.C., and R.D. Willig, "Economies of Scope," American Economic Review, 71:2, 1981, p. 268272.

Schneidewind, N.F., "The State of Software Maintenance," IEEE Transactions on Software Engineering, SE13(3), March 1987, p. 303310.

Skinner, W., "The Focused Factory," Harvard Business Review, 52:3, 1974, p. 113121.

- complete references are available upon request -